Application Specific Integrated Circuit for Electrical Instrumentation Processor

A Thesis Submitted

in Partial Fulfillment of the Requirements

for the Degree of

Master of Technology

by

Milind Parab

to the

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

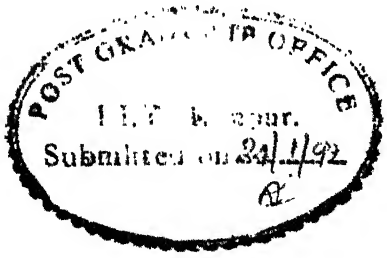
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

Jan 1997

CSE-1997-M-PRA-APP

CERTIFICATE

This is to certify that the work contained in the thesis entitled Application Specific Integrated Circuit for Electrical Instrumentation Processor by Milind Parab has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.



Dr. Rajat Moona,
Associate Professor,
Department of Computer Science & Engineerin
Indian Institute of Technology, Kanpur.

Acknowledgments

I wish to express my sincere gratitude to Dr. Moona for his invaluable guidance throughout the work. It was because of his enthusiastic efforts that I could complete this major design activity in such a short span. I also thank him for being my mentor and encouraging me in every possible way.

I wish to express my deepest sense of indebtness to my sponsorer, Dr. S.S. Sunderajan Chief Executive Officer, Datapro Electronics Pvt. Ltd., for providing me an opportunity to undertake this academically enriching experience.

I would like to thank Jayant Daftardar, Sr. Design Manager, Datapro Electronics Pvt. Ltd., for taking out time from his busy schedule to visit this place all the way from Pune and for his constant support and encouragement.

Special thanks to my friends Samir, Gokhale, Utpal, Warsi and Deepak for making my stay at IIT Kanpur an unforgettable and cherishable experience and to John and Suresh for providing emotional support which helped me in overcoming many problems.

I am grateful to my parents, brother & sister for serving as the constant source of motivation

Abstract

Dynamic watt-hour metering based on Faraday's principle has come a long way since their first inception some 80 years back. They are accurate, sturdy and reliable. The performance of dynamic meters, over the years has been satisfactory even in stringent environmental conditions. The limitations are of course, that they are easy to tamper with. Static watt-hour metering using the current state of the art discrete technology is not at all cost effective. The reasons are manifolds. For performance comparable to the dynamic meters, the electronics used must be highly fault tolerant. It is very difficult, if not impossible, to manufacture a discrete meter which would lie in the same cost bracket as that of dynamic meters, and would also comply to the applicable standards. The only solution lies in reducing the chip count which in turn would reduce the size and cost. Here, we propose a design of an ASIC for static watt-hour meters. It is a general purpose Integrated Circuit which can be used in any metering or similar electrical instrumentation applications. The advantages of using this are evident size reduction which in turn would reduce power consumption and cost, as very little peripheral support is required. To support any metering configuration we have upto 4 independent Voltage and Current channels. Meter performs computations for harmonic energy and total energy. Other features include full trivector computations and DC averaging. Direct pulse outputs and high sampling rate makes this ASIC an excellent choice for any metering application. The circuit was designed at gate level and was simulated in Verilog HDL. Total gate count is 10,300 gates excluding memory.

Contents

1 Introduction 1

1.1 Organization of the report 3

1.2 Motivation 3

1.3 Problems in dynamic metering. 5

1.4 Research at Datapro Electronics in the field of static watt-hour metering. 6

2 Requirement Analysis 8

2.1 Input requirements. 8

2.2 Computational requirements. 10

2.3 Output requirements. 10

2.4 Miscellaneous requirements 10

3 Design Specification. 12

4 Design 17

4.1 Data Acquisition Module (DAM) 17

4.1.1 DAM computations 19

4.1.2 DAM architecture complexity 23

4.1.3 Architecture 25

4.1.4 Control sequencer 26

4.2 Data Computation Module (DCM) 27

4.2.1 DCM computations 28

4.2.2 Power module architecture 33

4.2.3 Pulse outputs 36
4.3 Memory module 36

5 Future Work 39

A General Definitions 42
A.1 Definitions related to the functional elements 42
A.2 Definitions of meter quantities 43
A.3 Display of measured values. 44

B Control sequencer 45
B.1 Data acquisition module 45
B.2 Data computation module 50
B.2.1 Sequencer for phase computations 51
B.2.2 Control sequencer for system computations 54

C Memory 57
C.1 Filter intermediate data 57
C.2 Past filter voltage storage for sign of reactive power (R_{sign}) calculation 58
C.3 Squared and accumulated values 58
C.4 Accumulated powers 59
C.5 Trivector power and related parameters storage locations 59
C.6 System parameters and temporaries 60
C.7 Filter coefficients 61

D Accuracy requirements 62
D.1 Limits of error due to variation of current. 62
D.2 Influence quantities 63

References 66

Chapter 1

Introduction

In this thesis we design an ASIC for static polyphase trivector meters. The aim is to design a special purpose instrumentation processor which can be used mainly in applications like 3ϕ HT/LT metering, 1ϕ house hold metering. It can also be used in specialized applications like remote metering or prepaid metering or as a remote terminal unit to monitor online transformer loading and similar electrical instrumentation applications.

This project started with a background of research on similar lines conducted at Datapro Electronics, Pune. This work helped us to draw a clear specification for ASIC which was in line with the latest market requirements. The circuit description for the ASIC is written in Verilog HDL. Except for memory rest of the design is modeled using structural description.

The design of whole unit is divided in three main modules (*refer to figure 1.1*).

- Data Acquisition Module [DAM].
- Data Computation Module [DCM].
- Memory Module.

The modules DAM and DCM work in parallel. The DAM is the front end and it does the basic data crunching. A DAM cycle[1] is of 25μ seconds. In one DAM cycle a sample is read on analog interface, it is filtered, squared and finally accumulated

[1]The term cycle refers to period after which the computations would be repeated.

in the memory. The other computations in a DAM cycle are zero crossing detection, value adjustment, DC averaging and VI[2] computations. A slight variation in DAM computation is there for a voltage sample and a current sample. Results of DAM computations are stored in the memory, which can be read by DCM or an external processor. Other than checks for positive zero crossing all DAM computations are simple addition and multiplication. The hardware designed for this is based on a single multiplier accumulator unit with two parallel Booth multipliers.

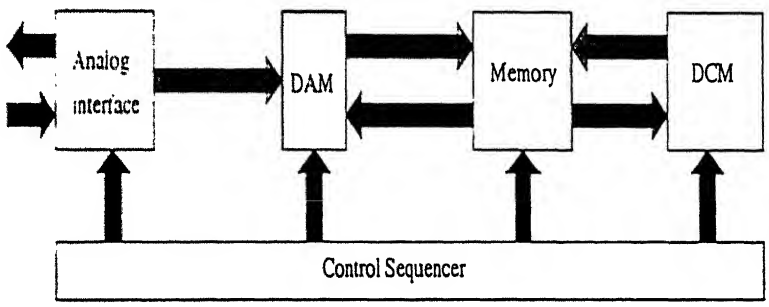


Figure 1.1: Block diagram

A DCM cycle is synchronized to the positive zero crossing. In approximately one second there is only one DCM cycle. DCM cycle time may vary depending on the zero crossings. The DCM reads the results of DAM computation from the memory and computes the trivector powers for all the phases[3]. It also calculates the pulse counts and program the corresponding pulse counter. There are direct pulse outputs corresponding to the trivector powers for each phase. The computations done in DCM are addition, subtraction, multiplication, division and square root. The hardware for DCM was optimized for space as there are very little computations to be done in a cycle period of one second.

The ASIC has 1K word (word is 16 bit wide) of on chip memory. Around 75% of this is used by DCM and DAM. The whole address space is also accessible on the external memory interface. Memory can be read in hold mode where the bus is granted to the peripheral requesting hold, or in asynchronous mode, where the memory read is permitted by ASIC in clock cycles when the memory is not accessed by either DAM or DCM.

[2]VI computation is the dot product of Voltage and Current curve.
[3]A pair of channel (Voltage and Current) is a phase.

1.1 Organization of the report

- This Chapter
  - Section §1.2 : Motivation for an industry sponsored project is normally market driven. In this section we would discuss the motivation of this project in brief.
  - Section §1.3 : Before static watt-hour meters came into the market dynamic meters were the only watt-hour meters available and these are still used in large numbers. This section discusses the exact problems with dynamic meters.
  - Section §1.4 : As we have said earlier much work has already been done at Datapro Electronics, Pune in the field of metering. This section discuss past eight years of work at Datapro.
- Other chapters
  - chapter §2 : This chapter lists the requirements which the ASIC should fulfill.
  - chapter §3 : The design of this ASIC is based on the specification which are drawn after a study of the current market requirements and the emerging trends. This chapter discusses the design specifications.
  - chapter §4 : The implementation of ASIC is discussed in this chapter.
  - chapter §5 : Design of this ASIC is the first step towards the manufacturing of the static watt-hour meter. Still there are a lot of issues which needs to be addressed. This chapter discusses the future work in this direction.

1.2 Motivation

The motivation for this project is the vast emerging market for static watt-hour[4] meters in India and abroad. To get an insight into the market we roughly see the

[4]See appendix for definitions.

requirements of the electricity boards in India, who are the major customers.

There are around 25 electricity boards in India. The bigger electricity boards like MSEB[5], APSEB, KSEB, RSEB, HSEB, OSEB has more than 10,000 big HT consumers. On an average if we consider 25 electricity boards with 5000 HT consumers. and if roughly we say a MTBF[6] of 3 years, and the cost per meter around 25,000 then the market offered by electricity boards for HT meters is of the order of 100 million[7] rupees per year.

Other than electricity boards there are institutions and cooperative societies who manage the power supply and distribution spanning 25 states in India. On top of this there are bodies like National Thermal Power Corporation who are constantly in the look for better technology for improving the infrastructure of distribution. All this together is a huge market for energy meters.

Much higher potential exists for LT meters and household meters. Then there are other areas like railway traction and portable metering which are new emerging markets.

With growing market the competition is on the rise mainly due to the entry of multinational companies and with their large market share worldwide, the result the cost is declining. What is required is a product which is accurate, reliable, sturdy, sleek and cost-effective. By reliability we mean that the meter should give a steady and repeatable performance. Since, the application ranges from small household users to large HT consumers, where the consumption is in tunes of millions of rupees per day, the product should be foolproof. Sturdy means that the meter should sustain environmental variations and should also be tamper proof.

From manufacturers point of view, to achieve the quality as demanded by the application, the manufacturing time processing should be minimum and it should be possible to automate the processing for large scale production. Further trimming or calibration of each individual meter should be avoided. The proposed ASIC would simplify the job of manufacturing a fault tolerant product at minimum cost.

[5]The last three letters stands for State Electricity Board & prefix is the name of the state.
[6]Refer Glossary.
[7]100 million $\Leftarrow \frac{5000 \times 25 \times 25000}{3}$

1.3 Problems in Dynamic metering.

Till date, most of the metering in India is done by dynamic meters. With growing sophistication in metering technology, the traditional dynamic meters are unable to cope up. The problems can be listed as follows :

- Flexibility.

  Since, the data is not digitized it is difficult to satisfy the market demands which are in pace with the current technology. Prepaid metering, remote billing are upcoming application areas where digitization is inevitable. It is difficult to make a dynamic meter which would operate for all four quadrants and would calculate all three vectors of power. Normally, one has to install separate meters for active energy, reactive energy and frequency. All this can be computed in a single unit if the voltage and current waveforms are digitized at the front end. In addition to the trivector computations features like monitoring load at some fixed interval of time, computation of energy in programmable time slots, harmonic energy computation, recording date and time of tamper events, recording the occurance of maximum demand with power factor and keeping record of energy and maximum demand for each month and similar applications are not possible in dynamic meters.

- Tampering.

  Tampering has been a problem since the inception of consumer supplier relationship. Since the dynamic meters do not store data it is difficult to trace out if at all some tampering was done and when was it done. Consumer may tamper the meter after the last reading is taken and may recover the connections before the next reading is being taken down. Low consumption is no indication of tampering. Exact empirical data as to when the tampering was done and how it was done is required to sue a consumer. Such recording is easily possible in static watt-hour meter.

Initial work towards computerization in the field of metering was digitization of the output data. The metering device was dynamic, based on Faraday's principle.

The number of rotation of the disc were digitized, and all further computations were based on this data. For digitization of disc rotation a simple approach is used such as the one described here.

The rotating disk is used in conjunction with opto-electronics. The optical path is blocked through the disk and gets open only when a slot cut in the disk passes through it. Thus one rotation yields one pulse on the opto-electronics. These pulses are counted for further processing. Earlier meters of this type were designed to work in controlled environment. This meters are generally costly and were meant mainly for laboratory testing, and are still used in standard test laboratories for calibration purposes. Such meter are commonly refered to as Rotary Static Meters (RSS meters).

1.4 Research at Datapro Electronics in the field of static watt-hour metering.

Since, this project is a joint venture of Datapro Electronics and IITK we feel it important to discuss the past work done by Datapro Electronics. Here we will discuss the brief history of developments in design of static watt-hour meters.

Research in the field of static energy meters was taken up by Datapro Electronics in the year 88-89. The very first version 'V1' was based on an 8 bit processor. The analog circuit used analog components such as RMS to DC converters and analog multipliers. Frequency was computed using frequency to voltage converters. Because, most of the computations were done using analog Integrated Circuits, high sampling was not required. Analog values thus computed were digitized for further processing, which were trivial in nature. However the analog computations could not provide the desired accuracy.

The next step was digitization of the data at the waveform level. It was using a lot of digital hardware and a very little analog counterpart. The major modification was a very high sampling rate. Other features included provision of a full functional keyboard display interface, communication, and printer interfaces.

Under controlled laboratory conditions the performance of this version was excellent. but since it used a lot of discrete components it was a costly system to manufacture and not very fault tolerant.

To cut down on digital hardware and hence on size and cost a better engineered meter was designed. This version 'V6' had been a major success. For the first time in this version, features like load monitoring and metering in time slots were incorporated.

Although version 'V6' had been stable it was not price competitive enough to dictate the market. The manufacturing process was shorter than the earlier versions but still not suitable for mass production. Unit to unit calibration remained a major bottleneck in mass production. This was the time when Datapro Electronics and IITK joined hands to develop an ASIC for static watt-hour meters.

Chapter 2

Requirement Analysis

ASIC application area is mainly static watt-hour meters of any configuration. Secondary applications are in remote terminal units and similar areas of electrical instrumentation. In this chapter we list out the general requirements of an ASIC catering to the above areas.

2.1 Input requirements.

- $3\phi 3W$ configuration.

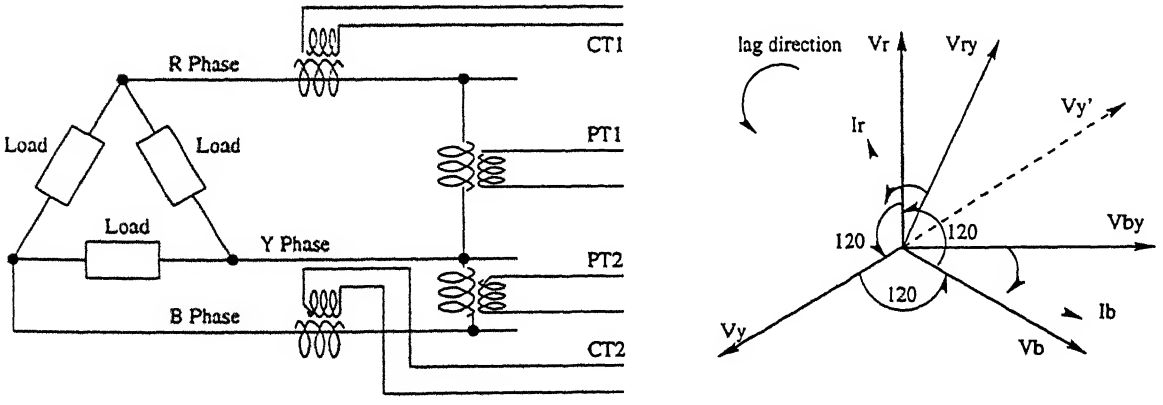


Figure 2.1: $3\phi 3W$ configuration

This is a delta configuration, normally seen in HT supplies. The meter is connected in two watt meter method. The two PT's and two CT's are connected

as shown in figure 2.1. Consider a supply[1] as V_r, V_y, V_b. The active power (W) is calculated as follows.

$$W_r = V_{ry} I_r \cos\phi_r; \quad W_b = V_{by} I_b \cos\phi_b$$

$$W_r = V_r I_r \cos(\phi_r + 30^o); \quad W_b = V_b I_b \cos(\phi_b - 30^o)$$

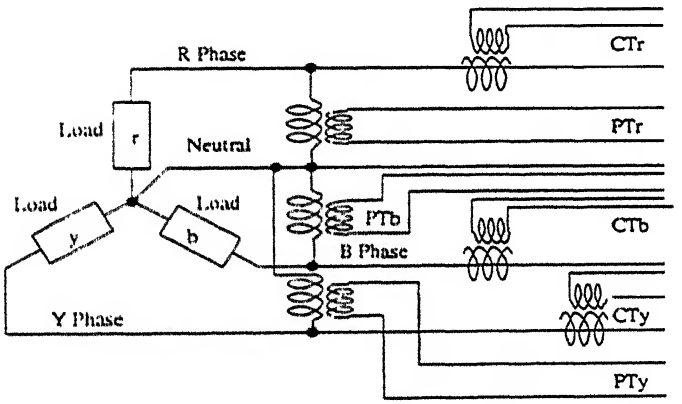
$$W_{total} = W_r + W_b$$

$$for \quad V = V_r = V_b; \quad I = I_r = I_b; \quad and \quad \phi = \phi_r = \phi_b$$

$$W_{total} = VI \ [\cos(\phi + 30^o) + \cos(\phi - 30^o)]$$

$$W_{total} = \sqrt{3} VI \cos\phi$$

- $3\phi 4$W configuration.



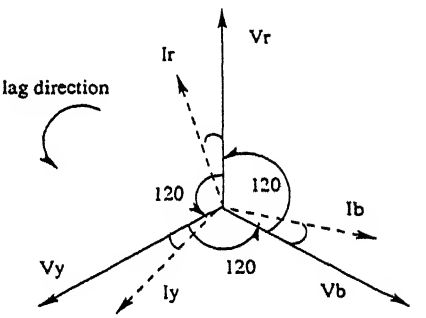


Figure 2.2: $3\phi 4$W configuration

This is star configuration, normally seen in LT supplies. The meter is connected in 3 watt meter method. The three PT's and three CT's are connected as shown in figure 2.2. Consider a supply as V_r, V_y, V_b. The active power (W) is calculated as follows.

$$W_r = V_{rn} I_r \cos\phi_r; \quad W_y = V_{yn} I_y \cos\phi_y; \quad W_b = V_{bn}.I_b \cos\phi_b$$

$$W_{total} = W_r + W_y + W_b$$

[1]Quantities repersented in capital letters are RMS quantities and those represented in small letters are instantaneous quantities.

$$for \quad V_{rn} = V_{yn} = V_{bn} = \frac{V}{\sqrt{3}}; \quad I = I_r = I_y = I_b; \quad and \quad \phi = \phi_r = \phi_y = \phi_b$$

$$W_{total} = \sqrt{3} V I \cos\phi$$

2.2 Computational requirements.

For a trivector meter, following are the minimum computations

- Active power W $\rightarrow VI\cos\phi$
- Reactive power R $\rightarrow VI\sin\phi$
- Apparent power S $\rightarrow \sqrt{W^2 + R^2}$

The above computations should be valid in all the 4 quadrants, that is phase angle can take value from 0° to 360°. The relavent IEC[2] standard, however specifies the range of operation from 0.5lag to 0.8lead i.e. 324° to 60°. Achieving such a wide dynamic range is clearly not possible in conventional meters.

2.3 Output requirements.

In its minimum configuration the meter should output pulses corresponding to the total watt-hour consumption. The constant of the pulse i.e. number of pulses per unit consumption of energy, may vary from meter to meter depending upon its rating. Also, it should be possible to select any of the computed trivector energy parameter for driving the pulse output. Further, it may be possible to display the computed trivector energy parameter and other relevant data on a display. Again the parameters displayed, their resolution, format or sequence may vary from meter to meter.

2.4 Miscellaneous requirements

- Harmonic energy computation.

[2]Refer appendix C for accuracy specifications

If possible the meter should compute the harmonic energy or should be able to segregate fundamental energy and total energy. As a minimum requirement the meter should be able to perform accurately even in the presence of 10% of third harmonic in current.

- Accuracy requirements.

  Meter should comply to the accuracy class of 0.2 S or 0.5 S as defined in IEC specifications for static watt-hour meters.

- Frequency computation.

  Normally a different frequency meter is used for measuring frequency. It is not a standard feature of dynamic meters. If possible frequency of the voltage waveforms should be computed.

  In polyphase configuration the meter works as long as one of the phase is active. The frequency may correspond to the active phase. If all the phases are active then the frequency may correspond to the first active phase in the sequence.

- DC averaging.

  In several electrical instrumentation applications like RTU it is required to compute the average value of DC inputs. For a single input RTU there should be at least one pin on which averaging is done. This may not be required in static watt-meters.

Chapter 3

Design Specification.

In this chapter we discuss the major design decisions and their basis. Implementation details can be found in the next chapter.

§1. 10 MHz clock.

In selecting the clock the aim was to start with the minimum clock speed, so as to minimize the power consumption (CMOS). As a minimum clock rate 10MHz clock was selected. At this clock rate other parameters like sampling speed and computational bandwidth were fixed. There were two constraints in designing the hardware to work at the selected clock speed. Firstly, it should be minimum i.e. it should not happen that we are using complicated digital hardware for the operations which otherwise could have been much simpler, had the clock been faster. Secondly, the hardware should work at a highest possible efficiency, ideally 100%. In other words, the utilization of the hardware should be maximum possible. The clock rate of 10MHz was found ideally suited for this hardware. The clock duty cycle is 50%.

§2. Eight input channel grouped as 4 pairs of VI channels or phases.

Minimum 3 phases are required to support any given three phase configuration. We support 1 extra phase so that the ASIC can be used in any specialized application like 4 single phase meters in a single box or 4 remote terminal

sensors etc. It was not possible to support more than 4 phases and moreover it was thought unnecessary as the major application area is metering.

§3. Five kilo Hertz sampling rate per channel.

There are two bounds on selection of sampling rates. The lower bound is given by Nyquist sampling theorem and the upper bound is system limitations. The aim is to select the maximum possible sampling rate which the system can support.

- Lower bound computation

  The frequency range of interest to us is upto 8th harmonic. The peak input operational frequency, fmax is 60Hz. 8th harmonic of fmax is 480 Hz. So the minimum sampling rate should be greater than 960 Hz.

- Upper bound computation

  The limiting factors are the conversion time of the ADC and the hardware data crunching speed. Typical conversion time of a successive approximation analog to digital converter is of the order of 20μsec. The computation period at 10MHz for one sample is around 25μsec i.e. at least 250 clock cycles are required for data crunching. So the permitted upper bound for a sample is 25μsec. Twenty five micro-seconds is equivalent to 40KHz of sampling rate. For 8 input channels, maximum allowed sampling frequency per channel would be 5KHz[1].

  960Hz $\ll$ sampling rate $\leq$ 5KHz. And the sampling rate selected was 5KHz.

§4. Sixteen bit analog input.

This resolution is required for a meter of accuracy class 0.2 S as shown below. The accuracy of the meter is measured from 1% to 120% of the rated load. For ease in calculation we shift the operational range from 0.833% to 100%.

[1] $8 \times 5KHz \Rightarrow 40KHz \Rightarrow \frac{1}{25\mu sec}$

- Now consider the case of 12 bit ADC.
  0.833% of the full scale count is $0.833 \times 4096/100 \Rightarrow 34$. Error of 1 bit in 34 $\Rightarrow$ 3% [not acceptable]
- Consider the case of 14 bit ADC.
  0.833% of the full scale count is $0.833 \times 16384/100 \Rightarrow 136$. Error of 1 bit in 136 $\Rightarrow$ 0.6% [not acceptable]
- consider the case of 16 bit ADC.
  Error of 1 bit in 544 $\Rightarrow$ 0.18% which is acceptable. In theory a 14.5 bit ADC would suffice since the error limit as prescribed in IEC 687, at 1% load for a 0.2S is $\pm$ 0.4%, but to have a safe limit we use a 16 bit ADC.

§5. Filtering.

For computation of harmonic parameters the input waveform is passed through a cascade series of 3 second order low pass IIR butterworth filter. Further, total and fundamental computations are done separately. Here, we discuss the frequency response of a 2nd order IIR filter which is the first filter in the set of three cascaded filters. The other two filters are the same except for slight modification of the coefficients.

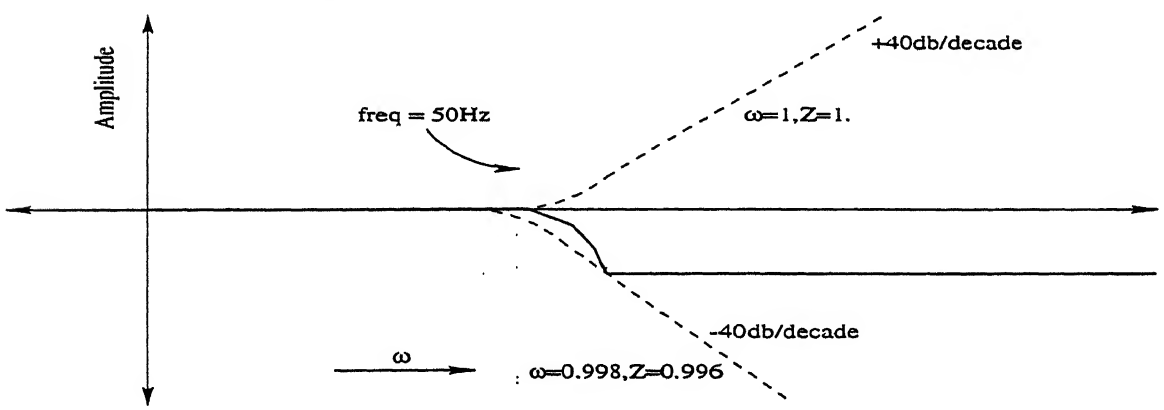


Figure 3.1: Filter response for stage 1

The coefficients of the first filter are $1.0, 2.0, 1.0, -1.9686, -1.9766$. The filter equation using the above coefficients is as follows :

$$y_n = x_n + 2.0x_{n-1} + x_{n-2} - 1.9686y_{n-1} - 1.9766y_{n-2}$$

where y_n is the output and x_n is the input sample. y_{n-1} and y_{n-2} are old output and old old output respectively, similarly x_{n-1} and x_{n-2} are old input and old old input.

This equation is for a 2^{nd} order low pass filter. Rewriting the above equation we get.

$$y_n + 1.9686y_{n-1} + 1.9766y_{n-2} = x_n + 2.0x_{n-1} + x_{n-2}$$

$$Y(z) + 1.9686z^{-1}Y(z) + 1.9766z^{-2}Y(z) = X(z) + 2.0z^{-1}X(z) + z^{-2}X(z)$$

$$Y(z)(1 + 1.9686z^{-1} + 0.9766z^{-2}) = X(z)(1 + 2z^{-1} + z^{-2})$$

$$\frac{Y(z)}{X(z)} = \frac{1 + 2z^{-1} + z^{-2}}{1 + 1.9686z^{-1} + 0.9766z^{-2}}$$

$$T(z) = \frac{z^2 + 2z + 1}{z^2 + 1.9686z + 0.9766}$$

The standard form for a filter is

$$1 + 2j\frac{\omega}{\omega_n}\zeta + [\frac{j\omega}{\omega_n}]^2$$

Rewriting in standard form

$$T(z) = \kappa\frac{(j\omega)^2 + 2j\omega + 1}{\frac{(j\omega)^2}{0.9766} + \frac{1.9686}{0.9766}j\omega + 1}$$

where κ is constant.

$$T(z) = \kappa\frac{1 + 2j\omega + (j\omega)^2}{1 + 2\frac{0.996}{0.988}j\omega + [\frac{j\omega}{0.988}]^2}$$

Comparing this equation with the standard form, we get
For numerator, $\zeta = 1, \omega_n = 1$, and for the denominator $\zeta = 0.996, \omega_n = 0.988$.
The filter response is as shown in figure 3.1.

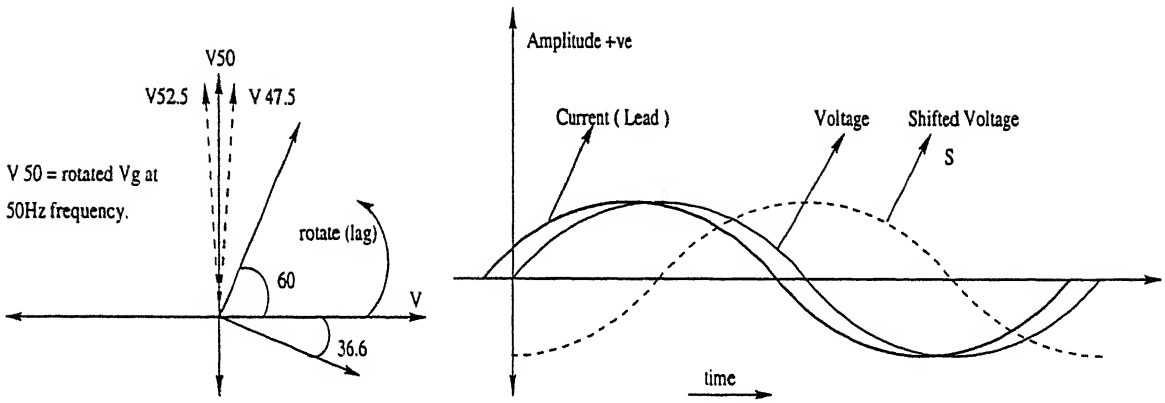


Figure 3.2: Block diagram

§6. Sign of reactive power.

For computing the sign of reactive power curve shifting is required. Theoretically, curve has to be shifted by exactly 90°. For a 50Hz periodic waveform, at 5KHz sampling rate 90° corresponds to 25 samples. With change in input frequency the shifting value would change. The range of operation of the static watt-hour meter as defined by IEC[2] is ±5% i.e. 47.5Hz to 52.5Hz. 90° shift at 47.5Hz is 26.3 samples and 90° shift at 52.5 Hz is 23.8 samples. This would introduce sign error in reactive power when the phase angle is near zero. It does not matter as the value of reactive power itself is close to zero at zero phase angle.

In the above calculation, we have assumed the fundamental (operational) frequency as 50Hz. The application may demand operation at frequency other than 50Hz. In such applications the number of samples by which voltage waveform is to be shifted, would vary. For this reason, the ASIC has the provision for programming the shift. The number of samples by which shifting is required can be programmed for any value from 0 to 31.

[2]IEC specification is there for meters at operational frequency equal to 50Hz or 60Hz.

Chapter 4

Design

Functionally, the ASIC is divided into three main modules. They are,

- Data Acquisition module [DAM].
- Data Computation module [DCM].
- Memory module.

DAM is the front end processor. It operates on the 16 bit digitized sample value and accumulates the results in the memory. DCM works on the data generated by DAM.

Both DAM and DCM do the computations in parallel. The memory area is mapped such that for each set of VI channels the DAM and DCM are working in non-overlapping memory areas termed as memory banks. The memory banks of DAM and DCM are switched after a duration of approximately one second, as will be seen later. The design of a module is discussed as follows. First, we would list the computations done, then we would discuss the architecture and finally we would describe the control sequencer.

4.1 Data Acquisition Module (DAM)

The results of DAM computations are stored in memory area, termed as memory bank. The switching of memory bank is done in synchronization with the input. This

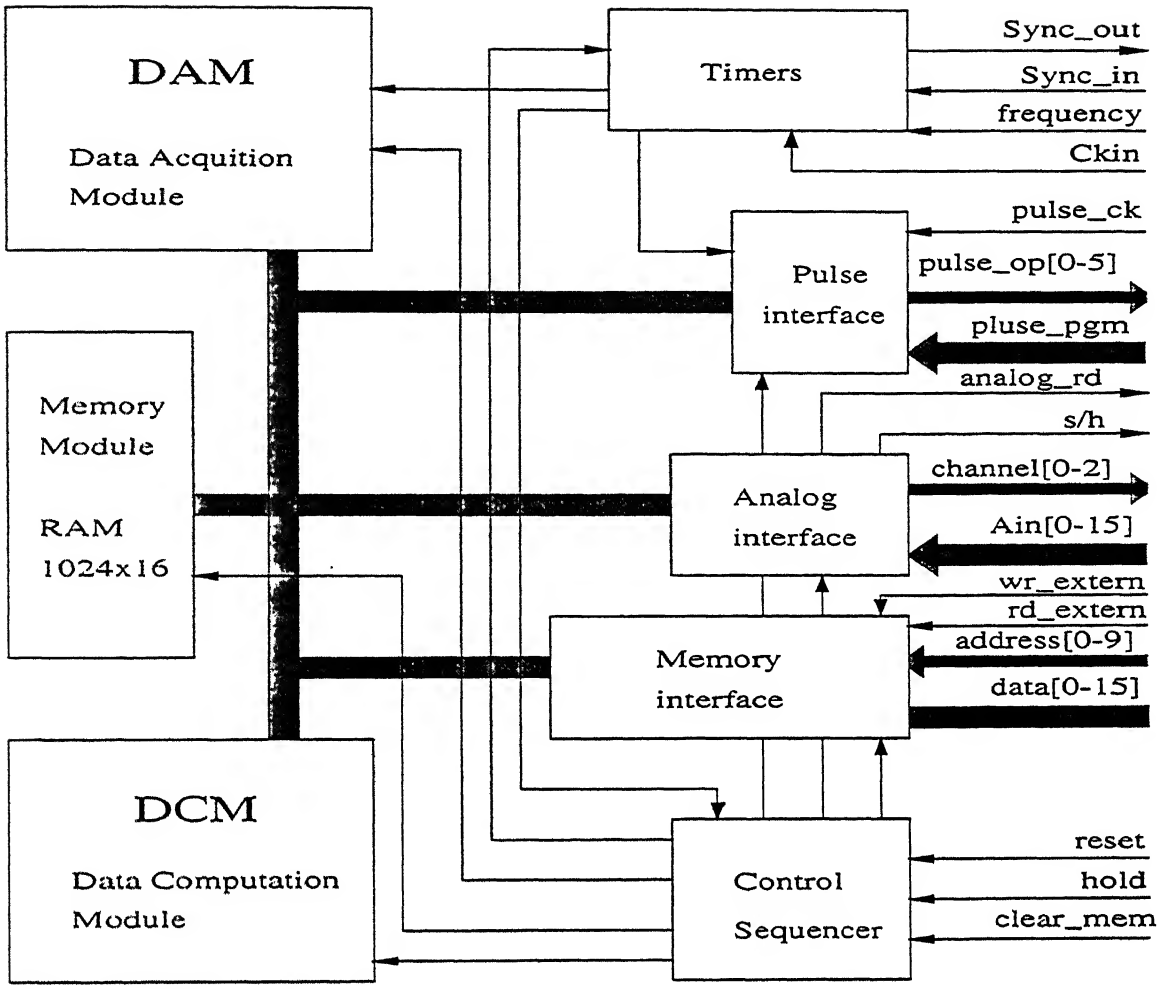


Figure 4.1: ASIC block diagram

synchronization is initiated by a positive trigger on *sync_in* pulse. The frequency of *sync_in* signal should be 1Hz. The memory banks are switched after occurance of first positive zero crossing in voltage channel, after trigger on *sync_in* pulse. If the input is DC then there would be no zero cross and the bank switching will not work. To avoid such a condition, if no positive zero crossing in voltage is detected for a duration of 100ms after trigger on *sync_in* then the bank is switched.

ASIC has 8 channels grouped as 4 pairs of voltage and current channels. On each channel 5000 samples are processed per second. So a DAM cycle is repeated after each 25μsec [1]. Since there are 8 channels, for a channel the sampling speed is 5KHz or one sample every 200μsec. Of the 8 channel 4 are voltage channels and 4

are current channels.

4.1.1 DAM computations

The computations are different for a voltage channel and a current channel. Following are the computations done in one DAM cycle for a voltage channel.

- Sampling.
- Zero crossing detection.
- Filtering.
- Squaring and Accumulation.
- DC Averaging.

Following are the computations done in one DAM cycle for a current channel.

- Sampling.
- Filtering.
- Squaring and Accumulation.
- VI computation.
- Sign of rVA computation.

§1. **Sampling**

Sampling comprises of two distinct tasks. One, is to read the value from ADC interface corresponding to a particular channel. Second, is to convert the read sample value from excess 8000 notation to 2'S complement form.

(a) Reading the sample value.
ASIC floats 3 bit channel number and simultaneously it asserts a *sample_hold* pulse. The pulse duration of *sample_hold* is 600nsec. The ADC

[1] $\frac{1}{5000 \times 8} = 25\mu$ seconds

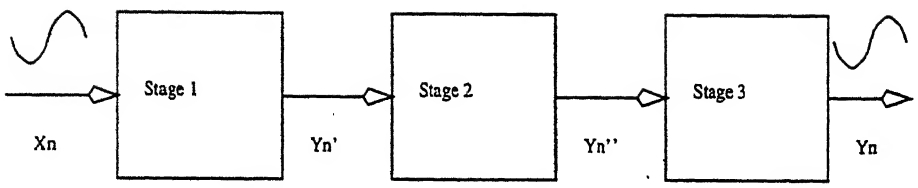


Figure 4.2: Block diagram

should start conversion with the falling edge of *sample_hold* pulse. For reading the converted data *analog_rd* pulse is asserted 23μsec after the falling edge of *sample_hold*. The external device should place data on Ain[0-15] pin within 50nsec (most ADCs meet these specifications) after the occurance of *analog_rd* pulse. The value read is stored in the memory.

(b) Sample value adjustment.

The sample value read at Ain[0-15] pins is in excess 8000_h notation i.e. 0 represents negative minimum value and $FFFF_h$ represents positive maximum value. The input value x_n in the range 0 to $FFFF_h$ is converted to 2'S complement notation as follows

$$x_n \Rightarrow x_n - 8000_h$$

$$if \ \ (x_n < 0) \ \ then \ \ x_n \ += 1$$

§2. Zero Cross Detection

The bank switching logic, is synchronized to the positive zero crossings in the input at voltage channel. A count of number of positive zero crossings detected on voltage channel after the last bank switch is computed and stored in the memory. Further, a count of number of samples between the two bank switch is computed. These counts are required in DCM for computations.

§3. Filtering

The 16 bit input sample x_n is passed through a pipelined series of three second order filters.

The filter equations are as follows :

$$y'_n = a_{00}x_n + a_{01}x_{n-1} + a_{02}x_{n-2} + b_{00}y'_{n-1} + b_{01}y'_{n-2}$$

$$x_{n-2} = x_{n-1}, \quad x_{n-1} = x_n, \quad y'_{n-2} = y'_{n-1}, \quad y'_{n-1} = y'_n;$$

$$y''_n = a_{10}y'_n + a_{11}y'_{n-1} + a_{12}y'_{n-2} + b_{10}y''_{n-1} + b_{11}y''_{n-2}$$

$$y'_{n-2} = y'_{n-1}, \quad y'_{n-1} = y'_n, \quad y''_{n-2} = y''_{n-1}, \quad y''_{n-1} = y''_n;$$

$$y_n = a_{20}y''_n + a_{21}y''_{n-1} + a_{22}y''_{n-2} + b_{20}y_{n-1} + b_{21}y_{n-2}$$

$$y''_{n-2} = y''_{n-1}, \quad y''_{n-1} = y''_n, \quad y_{n-2} = y_{n-1}, \quad y_{n-1} = y_n;$$

where y'_n, y''_n are the outputs of the intermediate stages. y'_{n-1}, y''_{n-1} are the old i.e. last cycle outputs of the intermediate stages. y'_{n-2}, y''_{n-2} are two cycle old outputs of the intermediate stages. Similarly, x_n, x_{n-1}, x_{n-2} are input, old input and old to old inputs respectively, and y_n, y_{n-1}, y_{n-2} are output, old output and old to old output respectively.

$a_{00}, a_{01}, a_{02}, b_{00}, b_{01}$ are first stage filter coefficients. Typical values are 1.0, 2.0, 1.0, -1.9686, -0.9766 for a 50Hz filter

$a_{10}, a_{11}, a_{12}, b_{10}, b_{11}$ are second stage filter coefficients. Typical values are 1.0, 2.0, 1.0, -1.9322, -0.9374 for a 50Hz filter

$a_{20}, a_{21}, a_{22}, b_{20}, b_{21}$ are third stage filter coefficients. Typical values are 1.0, 2.0, 1.0, -1.9130, -0.9154 for a 50Hz filter

All the filter coefficients are to be programmed in the memory at the time of power on.

§4. Squaring and Accumulation

For RMS value computation each 16 bit input sample x_n and its corresponding 16 bit filtered output y_n is squared and accumulated in the memory.

$$\sum_{i=1}^{n} x_n^2 \quad += x_n * x_n$$

$$\sum_{i=1}^{n} y_n^2 \quad += y_n * y_n$$

§5. VI computation

The 8 input channels are grouped into 4 pairs of voltage and current channels. If we consider a pair of channel which corresponds to a phase and name them as V and I, then the following computations are done.

$$\sum_{i=1}^{n} W_n \quad += v_n * i_n$$

Similarly for filtered voltage and filtered current.

$$\sum_{i=1}^{n} Wf_n \quad += v_{fn} * i_{fn}$$

where W_n , Wf_n are total and filtered active powers. v_n, i_n are voltage and current samples. v_{fn}, i_{fn} are filtered voltage and filtered current samples.

§6. Sign of reactive power R_{sign} computation

The voltage curve is shifted in lag direction by samples equivalent to $90^{\emptyset}$ at 50Hz. This is achieved as follows :

A circular queue of samples for all voltage channels is maintained. The length of the queue is equal to the count of number of samples by which shifting is to be done. There is a single queue pointer. The voltage sample value used for computation of sign is from the location pointed by the queue pointer. The current value of voltage sample is stored at the location pointed by queue pointer in the same cycle after the computations are over. Hence, the value used for sign computation is always old by a number of samples equal to length of the circular queue. All these computations are done on filtered value.

$$\sum_{i=1}^{n} Wf_n^{sft} \quad += v_{fn}^{sft} * i_{fn}$$

where Wf_n^{sft} active power value, the sign of which is the sign of rVA.
v_{fn}^{sft}, i_{fn} are shifted voltage and filtered current samples.

§7. DC averaging

The 16 bit input sample x_n is accumulated as follows.

$$\sum_{i=1}^{n} x_n \quad += x_n$$

The DAM architecture does not support direct accumulation because, what is designed is a pipelined multiplier-accumulator unit. Hence, for DC averaging the sample value is multiplied by 1 and accumulated. The value 1 is hardwired.

4.1.2 DAM Architecture complexity

Number of computations in DAM are.

- Total number of multiplications are 21. Out of this 21 multiplication 15 are required for filtering, 2 in squaring and accumulation, 1 in sign of reactive power computation and 1 for DC averaging.
- Total number of additions are 18. Out of this 18 additions 12 are required for filtering, 2 in squaring and accumulation, 2 in VI computations, 1 in sign of reactive power computation and one in DC averaging.
- Other computations are zero crossing detection, sample count, sample value adjustment, circular queue storage for sign of reactive power and shifting of data in filtering.

For multiplication the complexity is

- loading of operands requiring 2 cycles (2, 16 bit operands)
- storing of operands requiring 2 cycles (32 bit result)
- operation 16 cycles (shifting and adding)
- others, 2 cycles (clear and start).

Total 22 cycles.
For addition the complexity is

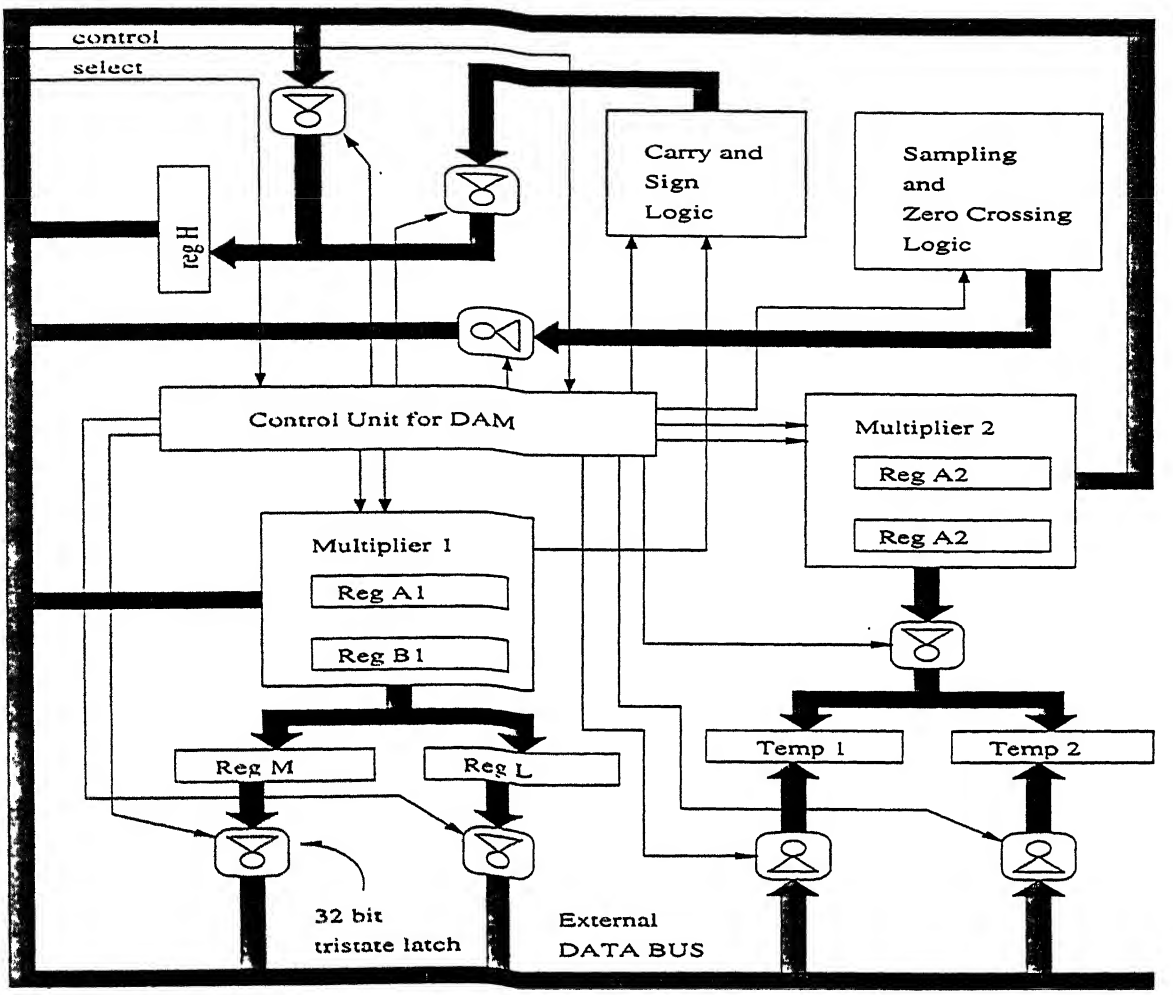


Figure 4.3: DAM Block Diagram

- loading of operands requiring 5 cycles (one 48 bit operand and two 32 bit operands)
- storing of operands requiring 3 cycles (48 bit result)
- operation 1 cycle.
- other 2 cycles (clear and start)

Total 11 cycles.

Total complexity 660 [2] clocks + other computations.

[2] 21x22+11x18

4.1.3 Architecture

The computations were optimized by combining multiplication and accumulation (addition) in one integral operation named as ***MAC*** (refer to figure 4.3). By doing so the extra load and store operations are saved. This fast multiplication module is build using two **radix 4 Booth** multipliers computing in parallel. The results of individual multiplications are added and stored in the accumulation register. Control word for the two parallel multiplier is common. There has to be a delay of at least three clock cycles between the initiation of the two multiplication commands. This mandatory delay is used for loading the operands and, is used to accumulate the results of the first multiplier in the accumulator. The results of the second multiplier are subsequently accumulated. In this way for a long sequence of multiplication and addition the effective delay comes out to be 7 clock cycles. For e.g. if the equation to compute is as follows

$$y = a_1x_1 + a_2x_2 + a_3x_3 + \cdots + a_nx_n$$

The computation of the above equation in multiplier-accumulator unit is

§1. **Load** a_1, x_1 in *two* cycles.

§2. **Sign extend & Start** multiplier one in *two* cycles.

§3. **Load** a_2, x_2 in *two* cycles.

§4. **Sign extend & Start** multiplier two in *two* cycles.

§5. Delay for 6 cycles.

§6. if all multiplications are not over, then Goto step 1.

Following the above sequence for implementing the filter equation of 4 addition and 5 multiplication we first clear the module and then load the multiplier in sequence defined above, without the need for accumulation. The shifted results from the accumulation register are read only once after the fifth multiplication in the filter equation is over.

The architecture of DAM has the following registers.

- For multiplier 1, Two 16 bit registers named as reg A1 and reg B1.
- For multiplier 2, Two 16 bit registers named as reg A2 and reg B2.
- For accumulator, 48 Bit accumulator register named as Acc H, Acc M and Acc L.

For executing the operations discussed above the DAM architecture supports the following instructions.

clear, load A1, load A1, sign_ext_mul1, mul1

load A2, load A2, sign_ext_mul2, mul2

load Accm H, load Accm M, load Accm L, read Acc H, read Acc M, read Acc L

load Z, Val Adjust, read Z, ZCD, incr

Each of the above instruction code is of 8 bits. For eg. code for **load A1** is 001 00 000. The decoding of this code is as follows (*from MSB*) :
First 3 bits is module select, next 2 bits are type of instruction and final 3 bits are instruction number.

4.1.4 Control Sequencer

The control sequencer [3] is built using simple and-or logic. All the operations in the DAM, listed above are executed within $25\mu sec$

The micro sequence for a single squaring and accumulation operation is as shown in the table below

[3] Refer appendix B for details

multiplier 1		Accumulator	
ck cycle	instruction	ck cycle	instruction
1	*clear*	2	*load Acc H*
5	*load A1 & B1*	3	*load Acc M*
6	*Sign ext A1*	4	*load Acc L*
7	*MAC*	18	*read Acc H*
-		19	*read Acc M*
-		20	*read Acc L*

The results are read from Accm reg A,B,C after a gap of 10 cycles. *Note : the results of squaring and accumulation is 43 bits*

$$(15 + sign) \times 2 + 13 \Rightarrow 43 + sign$$

In the above calculation, 30 bits are due to squaring and 13 bits are due to accumulation (assuming 2^{13} samples per DAM cycle).

The micro sequence for VI calculation is same as above except that the loading of reg A1 and reg B1 are in different clock cycles.

The micro sequence for value adjustment, zcd and sample count is as follows

ck	instruction	ck	instruction	ck	instruction
1	*load Reg Z*	4	*load Reg Z*	7	*load Reg Z*
2	*Val Adjust*	5	*ZCD*	8	*incr*
3	*read Reg Z*	6	*read Reg Z*	9	*read Reg Z*

4.2 The Data Computation Module (DCM)

This module calculates the trivector power and compute the pulse count for each of the power quantity. The pulses are programmed in a pulse counter which is clocked by an external clock. For trivector computation the values stored in memory by DAM are processed further. All the arithmetic in DCM is fixed point. As we have said earlier that the meter should work accurately from 1% to 120% of load. At low load the ADC count of the sample is low and hence the accumulated value would

be low. Hence at low loads, we shift the operands before the operation, whenever required. Similarly, check is performed to ensure that the results do not overflow at full load.

4.2.1 DCM computations

§1. Active power (W) computation

$$W \Rightarrow \sum_{i=1}^{n} \frac{V \times I}{n}$$

Full scale RMS values of V and I can be represented by Max value equal to 23170 (for 16 bit sample resolution peak RMS value is equal to $\frac{32768}{\sqrt{2}}$). Minimum value of W, is at 1% load and 90% voltage, which amounts to (assuming accumulation period of exactly one second)

$$\frac{231 \times 20853 \times 5000}{5000}$$

The numerator of the above equation is the accumulated value computed by DAM. Error of 1 bit in above division is negligible, hence no shifting is required. Maximum possible values of V and I are 32767 at constant and maximum (DC) load. At this load, the W value amounts to

$$\frac{(\pm 32767)^2 \times 5000}{5000}$$

Thus the accumulator for W, which stores the numerator should be at least 43 bits. The denominator which stores the sample count should be at least 13 bits. Therefore, for this division, we choose to have a hardware that can perform 48 / 16 bit division

§2. Sign extraction of W.

The division hardware in DCM is designed to handle unsigned numbers. The quantity W is a signed value. The sign of W is stored in the memory and for any further computation involving W only the magnitude of W is used. For doing this we take 2'S complement of W only if the value is negative. Therefore, the DCM architecture requires one 48 bit 2'S complement.

§3. RMS voltage or RMS current value (V, I) computation

Let the input sample of voltage or current be x_n Let the Voltage or Current value computed by DAM be X, where

$$X \Rightarrow \sum_{i=1}^{n} x_n \times x_n$$

The range of i is $5000 \leq i \leq 5500$. This is so because the bank switching logic is initiated exactly after 5000 samples and if no positive zero cross is detected by the end of 5500 samples then bank is forcibly switched. This is done so because, the input may be pure DC. The maximum number of accumulation i.e. 5500 corresponds to 13 bits. The word length of X is

$$15 \times 2 + sign + 13 \Rightarrow 43 + sign$$

We store this quantity using a 48 bit registers. The voltage or current computation is done in two steps, as follows

(a) The quantity X is divided by n, let the result be Y. Since n is atleast 5000 i.e. 13 bits wide, the bit width required to store Y is

$$43 - 13 + sign \Rightarrow 30 + sign$$

We use 32 bits representation to store this value.

(b) Square root of Y is the required RMS value of Voltage or Current.

$$V, I \Rightarrow \sqrt{Y}$$

The word length of V, I[4] is 32 bits. Here, a single bit error in square root would introduce upto 0.6% error[5] in the value of V or I. Since it is high, we multiply Y by 2^{16} before taking the square root. The square root is then taken for 48 bit (32+16) number which results in a 24 bit number. We store this as a 32 bit number.

[4]RMS values are represented in capital letters, whereas instantaneous values are represented in small letters.

[5]Minimum value of sample x_n at which accuracy is expected is 163 (1% of full scale). So the error is error of 1 bit in 163 which is equal to 0.6%

To perform both of these, we use a division circuit of 48 bit by 16 and a 48 bit square rooter.

§4. Apparent power (VI) computation

After computing RMS voltage and current values, apparent power is computed simply as $V \times I$. The operand length of V or I is 24 bits. The result of this computation before storage is shifted right by 16 bits.

$$S \Rightarrow \frac{V \times I}{2^{16}}$$

Since the values V or I were multiplied by 2^8 in the previous step, S is yield correct after division. S, can be stored in 32 bits (24+24-16).

§5. $\cos\phi$ computation

$$\cos\phi \Rightarrow \frac{W}{S}$$

where W and S are 32 bit wide. We cannot directly divide W by S in fixed point. A resolution of minimum 16 bits is required for $\cos\phi$ and hence W is shifted right by 16 bits before dividing. Maximum possible operand word length for division

$$\frac{32+16}{32}$$

To perform this we need a division circuit for 48 by 32 bit division. Output is stored in 16 bits as that much resolution is enough. Thus, the value represented is $c \times \cos\phi$ where, $c = 2^{16}$.

§6. $\sin\phi$ computation

$$c\sin\phi \Rightarrow \sqrt{c^2 - c^2 \times \cos^2\phi}$$

Where, c is 2^{16} (as discussed above). This computation is done in four steps as shown below.

(a) Squaring. Let $X \Leftarrow c\cos\phi \times c\cos\phi$. The operand length is 16 bit. The result of this multiplication is stored as a 32 bit unsigned number.

(b) Substraction. Let $Y \Leftarrow c^2 - X$. (c^2 is 2^{32} and is a constant). Y therefore, is just a 2'S complement of X. The result is again 32 bit unsigned number.

(c) Square root. Let $Z \Leftarrow \sqrt{Y}$. The result is 16 bit unsigned number.

(d) 2'S complement. After square root the results are unsigned. The sign of Z is the sign of R_{sign} where, R_{sign} is computed by DAM (for computation of R_{sign} refer to sign of reactive power computation in section 4.1.1). For this, we take 2'S complement of Z only if R_{sign} is negative.

To perform all the above computations we need circuit for 16 bit multiplication, 32 bit square root, and 32 bit 2'S complement.

§7. Reactive power (R) computation

R is calculated as $S \times c\sin\phi$ Where, c is equal to 2^{16}, resulting a number of width 48 bits. Since the results are scaled 2^{16}, the results are shifted right by 16 bits before storage, requiring a width of 32 bits. To perform this operation we need a 32 bit multiplication hardware.

§8. Average DC computation.

The DAM accumulates the samples received on 4 voltage channels. This quantity is averaged in DCM and stored as average DC value. Let, the accumulated value of DAM be X where,

$$X \Leftarrow \sum_{i=1}^{n} x_n$$

This value is divided by n in DCM, to get average DC value. The operand word length is 32 bit signed number, and the result word length is 16 bits. To perform this operation we need a 32 bit division hardware.

§9. Resetting of the accumulated value.

The memory is shared by DAM and DCM. The DCM works on the values computed by DAM and at the same time DAM computes a new set of values. Before the next bank switching occurs it is required to reset all the accumulated values stored in the current DCM bank to 0. Therefore, accumulated voltage, accumulated current, accumulated power, and average DC are all set to 0 at end of DCM computation.

§10. Per phase pulse output computation.

The DCM computes pulse count corresponding to all the trivector power parameters for all the phases. The powers are active power (W), reactive power (R) and apparent power (S). Of the three quantities, active power and reactive power are signed. Hence there are two pulse outputs corresponding to positive and negative value of the signed quantities. Therefore there are five pulse outputs per phase, namely pulse S, pulse W plus, pulse W minus, pulse R plus, pulse R minus. The following computations are done :

- pulse S = pulse S + S;
- if $W < 0$ then pulse W minus = pulse W minus - W;
  else pulse W plus = pulse W plus + W;
- if $R < 0$ then pulse R minus = pulse R minus - R;
  else pulse R plus = pulse R plus + R;

Of these quantities, upper 16 bits are loaded to the pulse counters (clocked by external clocks) and are set to 0 in pulse registers.

§11. System power computation

$$W_{sys} \Rightarrow W_1 + W_2 + W_3 + W_4$$

The storage of individual active powers is in 32 bits. The result of the above addition is shifted right by two bit and is stored as a 32 bit signed number. For performing this computation we need circuit for 32 bit addition.

§12. System pulse outputs.

Pulse outputs are computed corresponding to positive and negative values of system active power (W_{sys}). The logic for computation is same as that defined in per phase pulse output computation. For performing this computation we need circuit for 32 bit addition.

§13. Reseting of accumulated quantities.

The sequence of computation in DCM is

(a) Phase computations for all the phases.

(b) Filtered phase computations for all the phases.

(c) System computations for system active power.

(d) Computations for filtered system active power.

The DAM computed parameters are to be reseted before the next bank switching. This is done in two batches. The first batch of reseting is after filtered phase parameter computations as seen above. All the parameters cannot be reseted there as a few are required in later computations. The remaining parameters are reseted at the end of all computations. these additional parameters are sign of R, number of samples and zero crossing counts.

4.2.2 Power module architecture

Looking at the above requirements we could conclude that the energy module architecture should be capable of computing 32 $\times$ 32 bit multiplication, 48 / 32 bit division, 32 + 32 bit addition/subtraction and 48 bit square root. The approximated total computational bandwidth is 3058 cycles/operation, which is very small compared to the cycles available. So there is no timing constrains and hence this architecture is optimized for space *(refer figure 4.4)* .

There are 4, 32 bit registers namely A, B, AA, BB. The sequence of steps to be followed for each of the arithmetic operation are described below.

- Multiplication.
  The DCM supports 32 bit multiplication. The multiplication algorithm is **Booth** multiplication. Operand 1 is loaded in register A in two clock cycles. Operand 2 is loaded in register BB in two clock cycles. Two clock cycles are required for loading since word length of registers is 32 bits. For a 32 bit multiplication 64 clock cycles are required. The word length of the result is 64 bits and is stored in B and BB registers. The most significant word is in register B and the least significant word is in register BB. The 64 bit result is read from B and BB registers in four clock cycles.

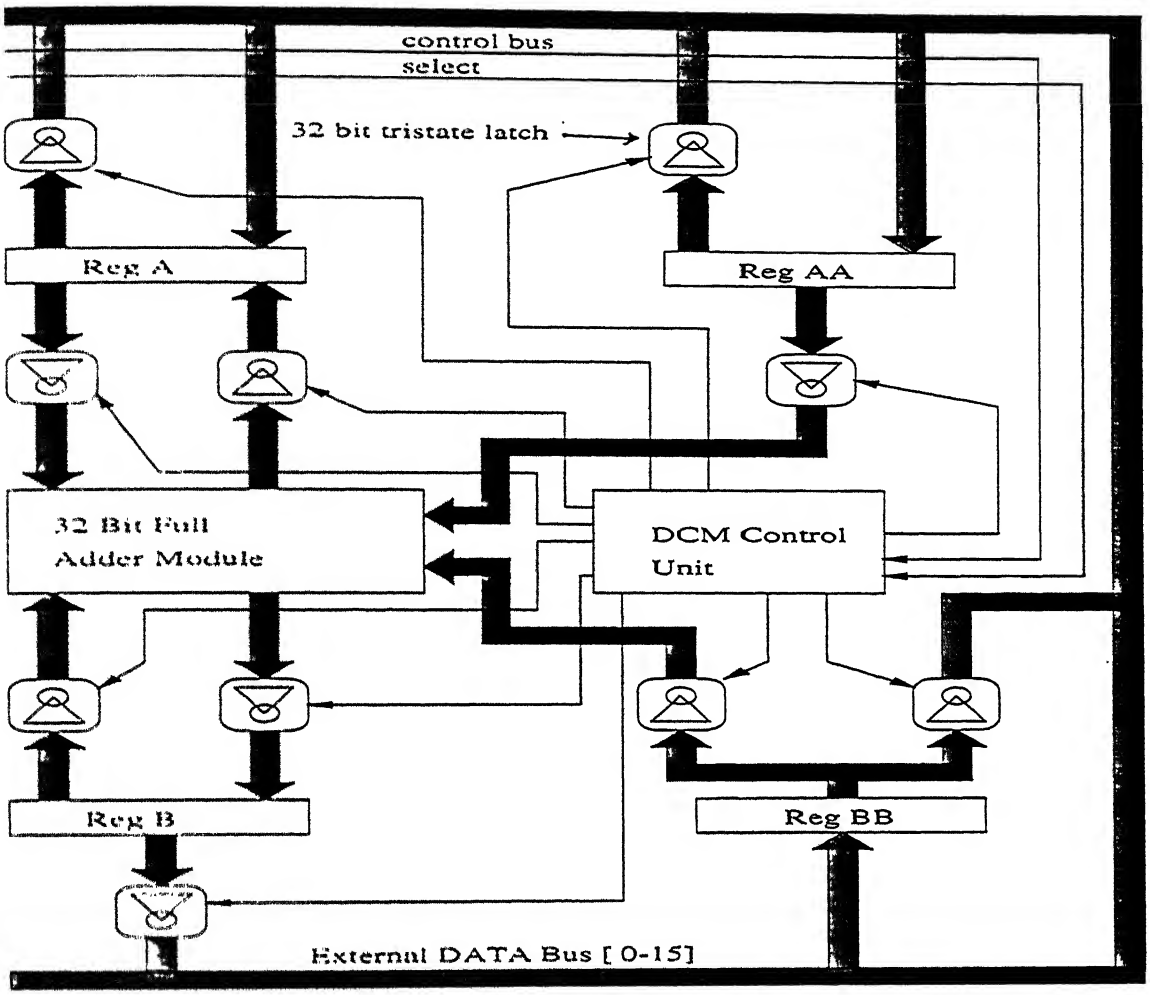


Figure 4.4: DCM block diagram

- Division.

  The DCM supports 63 by 32 bit division. The division algorithm is non-restoring division. Divisor is loaded in A reg and dividend in AA,BB pair. For dividend the most significant word is in register AA. Result of the division (in all the division cases discussed above) is less than or equal to 32 bits. Number of clock cycles required to do a division operation is 130, after which 32 bit results are available in BB register.

  In DCM computations it may happen that the divisor loaded is zero. Under such circumstances the results are undefined. To avoid such conditions, we shift zero in the result register if the divisor is zero. This is allright for our

computations as zero divisor will happen only in the first DCM computation cycle after a memory reset.

- Square root

  The DCM supports 64 bit square root. for this, 64 bit operand is loaded in register AA, BB pair in four clock cycles. The most significant word is loaded in AA register. After 64 clock cycles 32 bit result is available in register A. Two additional clock cycles are required for reading the results.

- Addition

  for addition, the first operand can be 64 bit wide while the second operand is restricted to 32 bits. First operand is loaded in AA,BB register pair in 4 clock cycle . Second operand is loaded in register A in two clock cycles. Addition itself requires two clock cycles after which 64 bit results is read from A,B register pair in 4 clock cycles.

The micro sequence for each operation is shown in the table below.

multiplication		Division		square root		Addition	
ck	instruction	ck	instruction	ck	instruction	ck	instruction
1	*clear*	1	*clear*	1	*clear*	1	*clear*
2	*load Al*	2	*load Al*	2	*load AAl*	2	*load Al*
3	*load Ah*	3	*load Ah*	3	*load AAh*	3	*load Ah*
4	*load BBl*	4	*load AAl*	4	*load BBl*	4	*load AAl*
5	*load BBh*	5	*load AAh*	5	*load BBh*	5	*load AAh*
6	*multiply*	6	*load BBl*	6	*square root*	6	*load BBl*
-		7	*load BBh*	-		7	*load BBh*
-		8	*divide*	-		8	*add*
71	*read Bl*	139	*read BBl*	71	*read Al*	11	*read Al*
72	*read Bh*	140	*read BBh*	72	*read Ah*	12	*read Ah*
73	*read BBl*	-		73	*read BBl*	13	*read Bl*
74	*read BBh*	-		74	*read BBh*	14	*read Bh*

4.2.3 Pulse outputs

As discussed earlier each phase has five (pulse S, pulse W plus, pulse W minus, pulse R plus, pulse R minus) pulse outputs. In addition two pulse outputs for W_{sys} are provided making a total of 22 pulse outputs. The resolution of a pulse count parameter is 32 bits. Out of the 32 bit we give out pulses corresponding to the 16 most significant bits. The residual 16 bits are accumulated in the next cycle of DCM computations. So, for a pulse output on the pin we need only a 16 Bit down counter. We support a total of six pulse output pins. For each of the pulse output pin there is one pulse output counter making a total of six down counters. Out of this two are direct pulse output which corresponds to the system active power. Rest of the four pulse output can be programmed to any of the 20 pulse outputs power parameters. Direct pulse outputs are given for system active parameters because these are the parameters for which the pulse outputs are generally desired. For programming a pulse output pin to a phase power parameter each pin has corresponding 5 address inputs. The pulse output counters are programmed to the corresponding power parameter. An additional divide by n counter may be used on the pulse output pin to support a pulse frequency for the electro-mechanical counters.

4.3 Memory module

Total memory area used by DCM and DAM is 770 words. The size of memory is 1024 words, which means that 254 locations are free. The various operations on the memory are as follows.

- Clearing the memory on power on reset

  The memory may contain garbage at power on. At power on it is required to clear all the locations for correct operation. To simplify the task of clearing 1024 locations, we provide a signal named *clear_mem.* this should be activated at the time of power on reset and should remain active at least for 200μsec to ensure that whole memory is cleared.

To achieve this we have a 10 bit address counter. This counter starts counting when *clear_mem* is active. The value zero to be written in memory for clearing is hardwired.

- Hold interface

  If hold is asserted then all the internal computations are frozen and the bus is granted to the peripheral requesting hold. There is no acknowledge corresponding to hold. The peripheral device after requesting hold should wait for a minimum duration of 500$n$sec before starting read/write operations. This interface can be used for programming filter constants.

  When hold is asserted, the bus is granted to the peripheral requesting hold only when both DCM and DAM are not using memory interface. A write from peripheral before the internal memory cycles are free would be ignored. That is the reason why we have given a safe limit of 500$n$sec before initiating actual writing in the memory.

  This is synchronous write i.e. the number of writes depends on the number of clock cycles for which *wr_extern* was active. For writing n number of locations *wr_extern* need be activated only once and should remain active till all n locations are written. Writing of one location requires 100ns. The only care which should be taken while writing is that the data on *data_extern* bus and address on *add_extern* bus should be valid at positive edge of the clock.

- Asynchronous read and write.

  The results of computation at each stage in DAM and DCM are stored in memory. Any location in the memory can be accessed for read or write. Hold interface can be used for the same, but it is advisable not to use hold interface as it disturbs the internal computational cycles. Moreover, it is unnecessary to use hold interface for single read or write. Asynchronous read or write is slow but the internal operations do not get affected. Whenever read or write is activated without hold, it is registered as asynchronous read or write. The memory operation is done in free cycles and after the operation is over *rd_wr_req_over* signal is asserted to acknowledge the completion.

Reading of the computation results should be done preferably when the computations of DCM are over so that the results are consistent. For example it should not happen that R value read is the one calculated in the current cycle and S value read is the one computed in the last cycle, or for the same parameter the two words read are computed in two different cycles. This may happen if data read operation is done when DCM module is active. To avoid this *sync_out* pulse can be used. This pulse is asserted exactly at the end of one second interval and it remains high until the computations by DCM module are over.

Chapter 5

Future Work

With the design of this ASIC the digital part of a watt-hour meter is over, but still its a long way to manufacture a watt-hour meter to the guidelines set earlier. The grey areas are transducer, analog part, power supply and engineering.

§1. Transducers.

Static meters are generally transformer operated devices. There is a potential transformer (PT) which is connected across the load and a current transformer (CT) connected in series with the load. The requirement for calibration or linearization in the meter is mainly due to the non-linearity of the transformer. The effect is more prominent in case of current because the range of operation of meter is from 0.1% to 120% of rated current, whereas it is from -10% to +10% in case of rated voltage. Its nearly impossible to manufacture a CT which is linear from 0.1% to 120% of rated current. The other constraints are of course cost, size and weight. The areas in which one should look into is using voltage dividers for PT and hall effect devices for CT's Other then transformation one should also see that proper isolation is maintained. Isolation is easy in case of transformer operated devices, due to magnetic coupling, which may not be trivial in voltage divider circuits.

§2. Analog

For a 0.2S meter a 16 bit resolution analog to digital, conversion is required. The conversion cycle is of 25μsec. These two factors, resolution and speed,

makes the design of analog part a bit complex. Successive approximation ADC's are fast (of the order of 20μsec) but common resolution is 12 bit and they are costly. We may use a sigma delta converter, but it would require using separate converter for each channel. Optimum balance in cost and speed must be striked.

§3. Power supply.

This is the most important and the most intricate part of a static watt-hour meter. The complexity arises due to the stringent requirements. For exact specification refer to IEC 687 specification. Roughly, the specifications are

(a) operating range -30% to +30% of rated voltage.

(b) Power consumption of the meter should be less than 1VA/phase, that demands a very high efficiency from the power supply.

(c) The operation should be consistent even if there are influences in supply voltage or in presence of fast transients in the supply.

(d) Operation should be guaranteed as long as at least one of the phase is active in a polyphase meter.

(e) It should pass impulse, AC voltage test and DC resistance tests.

§4. Engineering

By engineering we mean optimum placement of transducers, analog and digital circuits, and power supply unit. The placement should be such that the whole assembly occupies less space, no influence of one module on other should be there, proper heat conduction should be there, wiring length should be minimum, and of course it should look good. Other than aesthetics and ease of manufacturing the meter should be engineered such that it passes the following tests.

- Test of climatic influences (Dry heat, cold, damp heat cyclic).
- Test of mechanical requirements such as vibration, dust & water penetration and resistance to heat/fire.

- Test of electromagnetic compatibility such as immunity to RF interference, immunity to electromagnetic HF-fields and immunity to electrostatic discharge.

Appendix A

General Definitions

The majority of the following definitions have been taken from International Electrotechnical Vocabulary (IEV).

- **Watt-hour meter.** Instrument intended to measure active energy by integrating active power with respect to time.
- **Static watt-hour meter.** Meter in which current and voltage act on solid state (electronic) elements to produce an output pulse frequency proportional to watt-hours.
- **Meter type.** Term used to define a particular design of meter, manufactured by one manufacturer, having
  - Similar metrological properties :
  - The same uniform construction of parts determining these properties. The type may have several values of rated current and reference voltage.

A.1 Definitions related to the functional elements

- **Measuring element.** Part of the meter which produces an output pulse frequency proportional to the energy.

- **Output devices.** Test output. Device which can be used for testing the meter. Operational indicator. Device which gives a visible signal of the operation of the meter.
- Memory. Elements which stores digital information..
- Display. Devices which displays the content(s) of (a) memory(ies).
- Register. Electromechanical or electronic device comprising both memory and display which stores and displays information. A single display may be used with multiple electronic memories to form multiple registers.
- **Current circuit.** Internal connections of the meter and part of the measuring element through which flows the current of the circuit to which the meter is connected. *Note : The current has no DC content.*
- **Voltage circuit.** Internal connections of the meter, part of the measuring element and power supply for the meter, if the meter is not supplied by an external power supply, supplied with the voltage of the circuit to which the meter is connected.

A.2 Definitions of meter quantities

- **Rated current**[1] (I_n). Value of current in accordance with which the relavent performance of a transformer operated meter is fixed.
- **Maximum current** (I_{max}). Highest value of current at which the meter purpose to meet the accuracy requirements of this standard.
- **Reference voltage** (U_n)). Value of the voltage in accordance with which the relevant performance of the meter is fixed.
- **Reference frequency.** Value of the frequency in accordance with which the relevant performance of the meter is fixed.

[1]The term "voltage' and 'current" indicates r.m.s. values unless otherwise specified

- **Class index.** Number which gives the limits of the permissible percentage error, for all values of current between 0.05 I_n and I_{max}, for unity power factor (and in the case of polyphase meters with balanced loads) when the meter is tested under reference conditions (including permitted tolerances on the reference values) as defined in IEC 687.

- **Influence quantity.** Any quantity, generally external to the meter, which may affect its working performance

- **Reference conditions.** Appropriate set of influence quantities and performance characteristics, with reference values, their tolerances and reference ranges, with respect to which the intrinsic error is specified.

A.3 Display of measured values.

The information can be shown either with an electromechanical register or an electronic display. In the case of an electronic display the corresponding non-volatile memory shall have a minimum retention time of four months.

In case of multiple values presented by a single display, it must be possible to display the content of all relevant memories. When displaying the memory, the identification of each tariff applied shall be possible. The active tariff shall be indicated.

When the meter is not energized, the electronic display need not be visible. The principal unit for the measured values shall be the kilowatt-hour(kWh) or megawatt-hour(MWh).

Appendix B

Control Sequencer

The control sequencer for DAM and DCM is based on and-or logic. The exact sequence of instructions hardwired for controlling the operation of DAM and DCM is being shown in the tabular form. The table should be treated as a flow of events, from left to right and from top to bottom. One box of the table lists all the operations done in that particular clock cycle.

For example, the instruction listed in the second box (which corresponds to the operations in second clock of a DCM cycle) is Wr1a which means load register A of multiplier 1. The memory operation is read and the address is given by the address of location x^0_{n-2}.

B.1 Data Acquisition Module

There are 250 clocks in one DAM cycle, and there are 23 instructions supported by the DAM. The DAM instructions are :

- **clear** : Clears registers of multiplier 1 & 2 and the accumulator register.
- **Wr1a,Wr1b** : Load A and B registers of multiplier 1.
- **Wr1** : Load A and B registers of multiplier 1 with same data (in one clock cycle).
- **Sign ext1** : Extend the sign of multiplier 1.

- **mul1** : Start multiplication 1.
- **Wr2a,Wr2b** : Load A and B registers of multiplier 2.
- **Wr2** : Load A and B registers of multiplier 2 with same data (in one clock cycle).
- **Sign ext2** : Extend the sign of multiplier 2.
- **mul2** : Start multiplication 2.
- **rd_L,rd_M,rd_H** : Read lower middle and higher order registers of 48 bit accumulator.
- **Wr_L,Wr_M,Wr_H** : Load lower middle and higher order registers of 48 bit accumulator.
- **WrZ_{15}** : Load 15 bits of ZCD register.
- **WrZ_{16}** : Load 16 bits of ZCD register.
- **rdZ_{15}** : Load 15 bits of ZCD register.
- **rdZ_{16}** : Load 16 bits of ZCD register.
- **incr** : Increment the ZCD register by one.
- **VA** : Convert the value in ZCD register from excess 8000 notation to 2'S complement form.
- **ZCD** : Detect positive zero crossing.

Next three pages lists the sequence of operations for one DAM cycle. This operations consumes 250 clock cycles or 25μ seconds. The operations listed below are superset of operations in Voltage channel and Current channel (detailed DAM operations for a channel can be found in section 4.1).

clear memRd x^0_{n-1} bankWr	Wr1a x^0_{n-2} memRd	Wr1b a_{02} memRd	Sign ext1 memWr x^0_{n-2} bankRd	mul1 memRd x^0_n bankWr	Wr2a x^0_{n-1} memRd	Wr2b a_{01} memRd
sign ext2 memWr x^0_{n-1} bankRd	mul2	delay of six Cycles	Wr1a x^0_n	Wr1b a_{00} memRd	sign ext1 memRd y^0_{n-1} bankWr	mul1 memRd
Wr2a y^0_{n-2} memRd	Wr2b b_{01} memRd	sign ext2 memWr y^0_{n-2} bankRd	mul2	delay of six Cycles	Wr1a y^0_{n-1} memRd	Wr1b b_{00} memRd
sign ext1	mul1	delay of ten Cycles	$rdACM_b$ y^0_{n-1} memWr	clear memRd x^1_{n-1} bankWr	Wr1a x^1_{n-2} memRd	Wr1b a_{12} memRd
sign ext1	mul1	Wr2a x^1_{n-1} memRd	Wr2b a_{11} memRd	Sign ext2 memWr x^1_{n-1} bankRd	mul2	delay of six cycles 58
Wr1a y^0_{n-1} memRd	Wr1b a_{10} memRd	sign ext1 memRd y^1_{n-1} bankWr	mul1	Wr2a y^1_{n-2} memRd bankRd	Wr2b b_{11} memRd	Sign ext2 memWr y^1_{n-1} bankRd

mul2	delay of six cycles 72	Wr1a y^1_{n-1} memRd	Wr1b b_{10} memRd	Sign ext2	mul1	delay of ten cycles 86
$rdACMb$ y^1_{n-1} memWr	clear memRd x^2_{n-1} bankWr	Wr1a x^2_{n-2} memRd	Wr1b a_{22} memRd	Sign ext1 memWr x^2_{n-2} bankRd	mul1 memRd y^1_{n-1} bankWr	Wr2a x^2_{n-1} memRd
Wr2b a_{21} memRd	Sign ext2 memWr x^2_{n-1} bankRd	mul2	delay of six cycles 102	Wr1a y^1_{n-1} memRd bankRd	Wr1b a_{20} memRd	Sign ext1 memRd y^2_{n-1} bankWr
mul1	Wr2a y^2_{n-2} memRd	Wr2b b_{21} memRd	Sign ext2 memWr y^2_{n-2} bankRd	mul2	delay of ten cycles 116	Wr1a y^2_{n-1} memRd
Wr1b b_{20} memRd	Sign ext1	mul1	delay of ten cycles 130	$rdACM_b$ y^2_{n-1} memWr	clear memRd y^2_{n-1} bankWr	Wr1 x^0_n memRd
Wr_L $\sum x_n a$ memRd	Wr_M $\sum x_n b$	Wr_H $\sum x_n c$.	Sign ext1 memWr y^2_{n-1} bankRd	mul1	delay of ten clocks 148	rd_L $\sum x_n a$ memWr .

rd_M $\sum x_n b$ memWr	rd_H $\sum x_n c$ memWr	clear	Wr1 y^2_{n-1} memRd	Wr_L $\sum y_n a$ memRd	Wr_M $\sum y_n b$ memRd	Wr_H $\sum y_n c$ memRd .
Sign ext1	mul1	delay of ten cks 168	rd_L $\sum y_n a$ memWr	rd_M $\sum y_n b$ memWr	rd_H $\sum y_n c$ memWr	clear
Wr1a x^0_n memRd	Wr1b x^0_{n+1} memRd	Wr_L $\sum VIa$ memRd	Wr_M $\sum VIb$ memRd	Wr_H $\sum VIc$ memRd	Sign ext1	mul1
delay of ten cks 189	rd_L $\sum VIa$ memWr	rd_M $\sum VIb$ memWr	rd_H $\sum VIc$ memWr	clear	Wr1a y^2_{n-1} memRd	Wr1b $y^2_{n-1}L$ memRd
Wr_L $\sum VI^f a$ memRd	Wr_M $\sum VI^f b$ memRd	Wr_H $\sum VI^f c$ memRd	Sign ext 1	mul1	delay of ten cks 210	rd_L $\sum VI_f a$ memWr
rd_M $\sum VI^f b$ memWr	rd_H $\sum VI^f c$ memWr	clear	Wr1a y^2_{n-1} memRd	Wr1b y^{2sft}_{n-1} memRd	Wr_L $\sum rVA_{sgn} a$ memRd	Wr_M $\sum rVA_{sgn} b$ memRd
Wrc $\sum rVA_{sgn} c$ memRd	sign ext1	mul1	delay of ten cks 231	rd_L $\sum rVA_{sgn} a$ memWr	rd_M $\sum rVA_{sgn} b$ memWr	rd_H $\sum rVA_{sgn} c$ memWr
WrZ_{16} Read_IO	VA	rdZ^0_{15} x^0_n memWr	WrZ_{15} memRd	ZCD memRd y^2_{n-1} bankWr	rdZ_{15} memWr	WrZ_{16} memRd
incr memWr y^{sft}_{n-1} bankRd	rdZ_{15} memWr					

B.2 Data Computation Module

The DCM supports 22 instructions. Number of operations to be performed on DCM are very less compared to the DAM, and for all the DCM operations the computation period is less than 25μsecs (which is the period of one DAM cycle). Hence, one DCM operation cycle, for simplicity is rounded to one DAM cycle. In other words, any new operation on DCM starts with the first clock cycle of DAM, irrespective of the finishing time of the last DCM operation. For example if we initiate a multiplication operation, then the results would be available in 64 clocks but the reading of the result and initiating the next operation will start only after 250 clocks or 25μseconds.

There are 22 instructions in DCM. They are as follows :

- **clear** : Clear all registers of DAM.
- **read Al/read Ah** : Read lower/upper 16 bits of register A.
- **read Bl/read Bh** : Read lower/upper 16 bits of register B.
- **read BBl/read BBh** : Read lower/upper 16 bits of register BB.
- **load Al/load Ah** : Load lower/upper 16 bits of register A.
- **load AAl/load AAh** : Load lower/upper 16 bits of register AA.
- **load BBl/load BBh** : Load lower/upper 16 bits of register BB.
- **rdsft_bl** : Read register B[16-1], that is one bit shifted right.
- **rdsft_bh** : Read register A[0],B[31-17], that is one bit of register A and 15 MSB bits of register B.
- **Set sign** : Set sign flag to the MSB bit of DATA bus.
- **Clear sign** : Set sign flag to the not of MSB bit of DATA bus.
- **multiply** : Start 32 bit multiplication.
- **div** : Start 64/32 bit division.

- **sq** : Start 64 bit square root.
- **add** : Start 64 + 32 bit addition.
- **2S** : Start 64 bit 2'S Complement.

Following tables lists the sequence of operation in a DCM cycle (detailed DCM operations can be found in section 4.2).

B.2.1 Sequencer for Phase Computations

clear	Set sign $\sum W_a$ memRd	load Al $\sum W_c$ memRd	load Ah $\sum W_b$ memRd	load AAL $\sum W_a$ memRd	2S	mod($\sum W$) sign extraction
read Bl $T_1 d$ memWr	read Bh $T_1 c$ memWr	read Al $T_1 b$ memWr	clear memWr	load BBL $T_1 d$ memWr	load BBh $T_1 c$ mwmWr	load AAl $T_1 b$ memWr
load Al n memRd	div	$\frac{\sum W}{n}$	read BBl W_b memWr	load BBh W_a memWr	clear	load BBl $\sum V^2 c$ memRd
load BBh $\sum V^2 b$ memRd	load AAl $\sum V^2 a$ memRd	load Al n memRd	div	$Y = \frac{\sum V^2}{n}$	read BBl V_b memWr	read BBh V_a memWr
clear	load BBh V_b memRd	load AAl V_a memRd	Sq	$\sqrt{Y}$	read Al V_b memWr	read Ah V_a memWr
clear	load BBl $\sum I^2 c$ memRd	load BBh $\sum I^2 b$ memRd	load AAl $\sum I^2 a$ memRd	load Al n memRd	div	$Y = \frac{\sum I^2}{n}$
read BBl I_b memWr	read BBh I_a memWr	clear	load BBh I_b memRd	load AAl I_a memRd	sq	$\sqrt{Y}$

read Al I_b memWr	read Ah I_a memWr	clear	load BBl V_b memRd	load BBh V_a memRd	load Al I_b memRd	load Ah I_a memRd
mul	$\frac{VI}{2^{16}}$	read BBh VI_b memWr	read Bl VI_a memWr	clear	load BBh W_b memRd	load AAl W_a memRd
load Al VI_b memRd	load Ah VI_a memRd	div	$/fracWVI$	read BBl $\cos\phi_b$ memWr	clear	load BBl $\cos\phi_b$ memRd
load Al $\sum V^2 b$ memRd	mul	$\cos^2\phi$	read BBl $\cos\phi_b$ memWr	read BBh $\cos\phi_a$ memWr	clear	load Al $\cos\phi_b$ memRd
load Ah $\cos\phi_a$ memRd	2S	$Y = 1 - \cos^2\phi$	read Bl $\cos\phi_b$ memWr	read Bh $\cos\phi_a$ memWr	clear	load Al $\cos\phi_b$ memRd
load BBh $\cos\phi_a$ memRd	sq	$\sqrt{Y}$	read Al $\cos\phi_b$ memWr	clear	set sign $sign_a$ memRd	load Al $\cos\phi_b$ memRd
2S	sign x $\sin\phi$	read Bl $\cos\phi_b$ memWr	clear	load BBl VI_b memRd	load BBh VI_a memRd	load Al $\sin\phi_b$ memRd
multiply	$VI\sin\phi$	read BBh rVA_b memWr	read Bl rVA_a memWr	clear	load BBl $\sum AV_b$ memRd	load BBh $\sum AV_a$ memRd
load Al n memRd	div $VI\sin\phi$	$\frac{AVDC}{n}$	read BBl $AVDC$ memWr	clear	set sign $\sum W_a$ memRd	load Al $\sum W_b$ memRd
load Ah W_a memRd	2S	$\pm W$	read Bl W_b memWr	read Bh W_a memWr		

clear	read Al $\sum V_c^2$ memWr	read Ah $\sum V_b^2$ memWr	read Bl $\sum V_a^2$ memWr	read Bh $\sum I_c^2$ memWr	read BBl $\sum I_b^2$ memWr	read BBh $\sum I_a^2$ memWr
	read Al $\sum W_c$ memWr	read Ah $\sum W_b$ memWr	read Bl $\sum W_a$ memWr	read Bh $\sum AVdc_b$ memWr	read BBl $\sum AVdc_a$ memWr	memWr
clear	clear sign W_a memRd	load Al W_b memRd	load Ah W_a memRd	load BBl W_b^{p+} memRd	add	
read Bl W_b^{p+} memWr	read Bh W_a^{p+} memWr	clear	set sign W_a memRd	load Al W_b memRd	load Ah W_a memRd	load BBl W_b^{p-} memRd
	read Bl W_b^{p-} memWr	read Bh W_a^{p-} memWr				
add		clear	clear Sign rVA_a memRd	load Al rVA_b memRd	load Ah rVA_a memRd	load BBl rVA_b^{p+} memRd
add		read Bl rVA_b^{p+} memWr	read Bh rVA_a^{p+} memWr	clear	set sign rVA_a memRd	load Al rVA_b memRd
load Ah rVA_a memRd	load BBl rVA_b^{p-} memRd	add		read Bl rVA_b^{p-} memWr	read Bh rVA_a^{p-} memWr	clear
load Al VI_b memRd	load Ah VI_a memRd	load BBl VI_b^p memRd	add		read Bl VI_b^p memWr	read Bh VI_a^p memWr

B.2.2 Control sequencer for System computations

clear	load BBl W_{1b} memRd	load BBh W_{1a} memRd	load Al W_{2b} memRd	load Ah W_{2a} memRd	add	
read Bl T_{1b} memWr	read Bh T_{1a} memWr	clear	load BBl W_{3b} memRd	load BBh W_{3a} memRd	load Al W_{4b} memRd	load Ah W_{4a} memRd
add		read Bl T_{2b} memWr	read Bh T_{2a} memWr	clear	load BBl T_{1b} memRd	load BBh T_{1a} memRd
load Al T_{2b} memRd	load Ah T_{2a} memRd	add		rdsft Bl W_{sb} memWr	rdsft Bh W_{sa} memWr	clear
load BBl rVA_{1b} memRd	load BBh rVA_{1a} memRd	load Al rVA_{2b} memRd	load Ah rVA_{2a} memRd	add		read Bl T_{1b} memWr
read Bh T_{1a} memWr	clear	load BBl rVA_{3b} memRd	load BBh rVA_{3a} memRd	load Al rVA_{4b} memRd	load Ah rVA_{4a} memRd	add
	read Bl T_{2b} memWr	read Bh T_{2a} memWr	clear	load BBl T_{1b} memRd	load BBh T_{1a} memRd	load Al T_{2b} memRd
load Ah T_{2a} memRd	add		rdsft Bl rVA_{sb} memWr	rdsft Bh rVA_{sa} memWr	clear	load BBl W_{sb} memRd
load BBh W_{sa} memRd	load Al W_{sb} memRd	load Ah W_{sa} memRd	multiply		read BBl T_{1d} memWr	read BBh T_{1c} memWr
read Bl T_{1b} memWr	read Bh T_{1a} memWr	clear	load BBl rVA_{sb} memRd	load BBh rVA_{sa} memRd	load Al rVA_{sb} memRd	load Ah rVA_{sa} memRd

multiply		read BBl T_{2d} memWr	read BBh T_{2c} memWr	read Bl T_{2b} memWr	read Bh T_{2a} memWr	clear
load BBl T_{1d} memRd	load BBh T_{1c} memRd	load AAl T_{1b} memRd	load AAh T_{1a} memRd	load Al T_{2d} memRd	load Ah T_{2c} memRd	add
	read Bl T_{1d} memWr	read Bh T_{1c} memWr	read Al T_{1b} memWr	read Ah T_{1a} memWr	clear	load BBl T_{1b} memRd
load BBh T_{1a} memRd	load Al T_{2b} memRd	load Ah T_{2a} memRd	add		read Bl T_{1b} memWr	read Bh T_{1a} memWr
clear	load BBl T_{1d} memRd	load BBh T_{1c} memRd	load AAl T_{1b} memRd	load AAh T_{1a} memRd	sq	
read Al VA_{sb} memWr	read Ah VA_{sa} memWr	clear	clear sign W_{sa} memRd	load Al W_{sb} memRd	load Ah W_{sa} memRd	load BBl W_{sb}^{p+} memRd
add		read Bl W_{sb}^{p+} memWr	read Bh W_{sa}^{p+} memWr	clear	set sign W_{sa} memRd	load Al W_{sb} memRd
load Ah W_{sa} memRd	load BBl W_{sb}^{p-} memRd	add	read Bl W_{sb}^{p-} memWr	read Bh W_{sa}^{p-} memWr	clear	clear sign rVA_{sa} memRd
load Al rVA_{sb} memRd	load Ah rVA_{sa} memRd	load BBl rVA_{sa}^{p+} memRd	add		read Bl rVA_{sb}^{p+} memWr	read Bh rVA_{sa}^{p+} memWr
clear	set sign rVA_{sa} memRd	load Al rVA_{sb} memRd	load Ah rVA_{sa} memRd	load BBl rVA_{sb}^{p-} memRd	add	

read Bl rVA^{p-}_{sb} memWr	read Bh rVA^{p-}_{sa} memWr	clear	load Al VI_{sb} memRd	load Ah VI_{sa} memRd	load BBl VI^{p}_{sb} memRd	add
	read Bl VI^{p}_{sb} memWr	read Bh VI^{p}_{sa} memWr	clear	read Al $sign_{1c}$ memWr	read Ah $sign_{1b}$ memWr	read Bl $sign_{1a}$ memWr
read Bh $sign_{2c}$ memWr	read BBl $sign_{2b}$ memWr	read BBh $sign_{2a}$ memWr	clear	reset rVAsign $\phi_{1,2}$	read Al $sign_{3c}$ memWr	read Ah $sign_{3b}$ memWr
read Bl $sign_{3a}$ memWr	read Bh $sign_{4c}$ memWr	read BBl $sign_{4b}$ memWr	read BBh $sign_{4a}$ memWr	reset rVAsign $\phi_{3,4}$	read Al n_1 memWr	read Ah n_2 memWr
read Bl n_3 memWr	read Bh n_4 memWr	read BBl zcd_1 memWr	read BBh zcd_2 memWr	rdsft Bl zcd_3 memWr	rdsft Bh zcd_4 memWr	reset n, and zcd

Appendix C

Memory

C.1 Filter intermediate data

Note : All offsets are in Hexadecimal

content	offset	content	offset	content	offset
x_n^0	000	x_{n-1}^0	001	x_{n-2}^0	002
y_{n-1}^0	003	y_{n-2}^0	004	x_{n-1}^1	005
x_{n-2}^1	006	y_{n-1}^1	007	y_{n-2}^1	008
x_{n-1}^2	009	x_{n-2}^2	00A	y_{n-1}^2	00B
y_{n-2}^2	00C				

Total memory usage =

$$16(words) * 8(channels) = 128 words.$$

For calculating the address of filter intermediate values use the following equation

$$address = offset + 16 * channel\_number.$$

Where, offset is given in the above table and channel number range is 0 to 7. For example, the address of y_{n-1}^0 for channel 4 is equal to

$$address = 003 + 40 = 043H$$

C.2 Past filter voltage storage for sign of reactive power (R_{sign}) calculation

$$address = 128 + 32 * channel\_number + count$$

Where the count can be 0 to 31. Total memory usage,

$$32(words) * 4\,(Voltage\_Channels) = 128\,words.$$

C.3 Squared and accumulated values

content	offset	content	offset
$\sum_{i=1}^{n} x_n^2$	100	$\sum_{i=1}^{n} y_n^2$	103
$\sum_{i=1}^{n} x_n$	106		

Total memory usage =

$$8words * 8\,channels * 2[filter/total] = 128\,words.$$

For calculating the address of squared and accumulated values use the following equation

$$address = offset + 64 * Bank + 8 * channel\_number.$$

where offset is given in above table. Bank can be either 0 or 1, and channel number range is 0 to 7. For example, the address of $\sum_{i=1}^{n} x_n$ for channel 4 and bank 1 is equal to

$$address = 106 + 40 + 20 = 166H$$

C.4 Accumulated powers

content	offset	content	offset
$\sum_{i=1}^{n} V * I$	180	$\sum_{i=1}^{n} V_f * I_f$	183
$\sum_{i=1}^{n} V_f^{shft} * I_f$	186	Zero cross count	189
Sample Count	18A		

Total memory usage =

$$16 words * 4 channel\_pairs * 2[filter/total] = 128 words.$$

For calculating the address of accumulated powers use the following calculation :

$$address = offset + 64 * Bank + 8 * channel\_number.$$

where offset is given in above table. Bank can be either 0 or 1, and channel number range is 0 to 7. For example, the address of $\sum_{i=1}^{n} V * I$ for channel 4 and bank 1 is equal to

$$address = 180 + 40 + 20 = 1E0H$$

C.5 Trivector power and related parameters storage locations

content	offset	content	offset
Voltage (V)	200	Current (I)	210
Angle	220	AV DC	230
Active pwr (W)	240	Pls cnt + (W)	250
Pls cnt - (W)	260	Ract pwr (R)	270
Pls cnt + (R)	280	Pls cnt - (R)	290
App pwr (S)	2A0	Pls cnt (S)	2B0

For calculating the address use the following equation

$$address = offset + 8 * filter/total + 2 * channel\_pair\_number.$$

where offset is given in above table. Filter/total is 0 for total values and 1 for filtered values, channel_pair range is 0 to 3. For example, the address of voltage for second phase *(filtered)* is equal to

$$address = 200 + 8 + 4 = 20CH$$

Each of the below set of parameters e.g Voltage (V) occupies 16 words. The size 16 is calculated as follows : Size of each value is 32 bit i.e. 2 words. There are total 4 pairs of channels, hence it makes $2 * 4 = 8$ words. Further the filtered and total values are calculated separately and that makes it 16 words.

C.6 System parameters and temporaries

For filtered parameters, the offset = offset + 2.

content	offset
Active Power System (Ws)	2C0
Pulse count +Ve (Ws)	2C4
Pulse count -Ve (Ws)	2C8
reactive Power System (R_s)	2CC
Pulse count +Ve (R_s)	2D0
Pulse count -Ve (R_s)	2D4
Apparent power system (S_s)	2D8
Pulse count (S_s)	2DC
T1 (Temporary loc)	2E0
T2 (Temporary loc)	2E4

C.7 Filter coefficients

The memory locations from address $2E8_h$ to address $3EF_h$ is unused.

data	offset	data	offset	data	offset
a_{00}	3F0	a_{01}	3F1	a_{02}	3F2
b_{00}	3F3	b_{01}	3F4	a_{10}	3F5
a_{11}	3F6	a_{12}	3F7	b_{10}	3F8
b_{11}	3F9	a_{20}	3FA	a_{21}	3FB
a_{22}	3FC	b_{20}	3FD	b_{22}	3FE

Appendix D

Accuracy requirements

D.1 Limits of error due to variation of current.

When the meter is under the reference conditions as specified in IEC 687, the percentage errors shall not exceed the limits for the relevant accuracy class given in table below If the meter is designed for the measurement of energy in both directions, the values given in the table below shall apply for each direction.

Value of current	Power factor	Error limits	
		0.2 S	0.5 S
0.01 $I_n \leq I < 0.05 I_n$	1	±0.04	± 1.0
0.05 $I_n \leq I \leq I_{max}$	1	±0.2	± 0.5
0.02 $I_n \leq I < 0.1 I_n$	0.5 lagging	±0.5	±1.0
	0.8 leading	±0.5	±1.0
0.1 $I_n \leq I \leq I_{max}$	0.5 lagging	±0.3	±0.6
	0.8 leading	±0.3	±0.6
When specially requested by the user 0.1 $I_n \leq I \leq I_{max}$	0.25 lagging	±0.5	±1.0
	0.5 leading	±0.5	±1.0

The difference between the percentage error when the meter is carrying a single-phase load and a balanced polyphase load at rated current and unity power factor,

shall not exceed 0.4% and 1.0% for meters of class 0.2 S and 0.5 S respectively.

D.2 Influence quantities

Limits of error due to influence quantities (Voltage variation, frequency variation, waveform, phase sequence, voltage unbalance) are listed in the following table. The additional percentage error due to the change of influence quantities with respect to reference conditions, shall not exceed the limits for the relevant accuracy class given in table below.

Influence Quantity	Load Balanced	Power Factor	limits of variation 0.2S	0.5S
Measuring circuit Voltage ±10%	0.05 $I_n \leq I \leq I_{max}$ 0.1 $I_n \leq I \leq I_{max}$	1 0.5 lagging	0.1 0.2	0.2 0.4
Frequency Variation ±5 %	0.05 $I_n \leq I \leq I_{max}$ 0.1 $I_n \leq I \leq I_{max}$	1 0.5 lagging	0.1 0.1	0.2 0.2
Wave-form: 10% of third harmonic in the current	$0.05I_n \leq I \leq I_{max}$	1	0.1	0.1
Reversed phase sequence	$0.1I_n$	1	0.05	0.1
Voltage unbalance	I_n	1	0.5	1.0
Auxiliary voltage ±15%	0.01 I_n	1	0.05	0.1
Phase of auxiliary supply voltage changed by ±120°	$0.01I_n$	1	0.1	0.2

Glossary

HT	High Tension.
LT	Low Tension.
RTU	Remote Terminal Unit.
CT	Current Transformer.
PT	Potential Transformer.
DAM	Data Acquition Module.
DCM	Data Computation Module.
FESP	Front End Signal Processing.
RESP	Rear End Signal Processing.
KWh	Kilo Watt-hour.
rkVAh	Reactive Kilo Watt-hour.
KVAh	Apparent Kilo Watt-hour.
RMS	Root Mean Square Value.
IIR	Infinite Input Response.
MTBF	Mean Time Between Failure.
RSS	Rotary Satic Meters.
IEC	International Electrotechnical Comission.
ASIC	Application Specific Integrated Circuit.
HDL	Hardware Description Language.
APSEB	Andra Pradesh State Electricity Board.
CESS	Co-operative Electric Supply Association.
KSEB	Karnataka State Electricity Board.
MSEB	Maharashtra State Electricity Board.
NTPC	National Thermal Power Corporation.

OSEB	Orissa State Electricity Board.
RSEB	Rajasthan State Electricity Board.
VSI	Vasantdada Sugar Institute.

CSE-1997-M-PRA-App